Technology Tarena 达内科技

PHP基础

主讲：Mr.King
Q Q：382771946
邮箱：382771946@qq.com
博客：http://www.phpfamily.org
微博：新浪微博@Mr_King_Hs

PHP介绍

知识讲解

PHP运行环境

LAMP:Linux+Apache+MySQL+PHP

WAMP:Windows+Apache+MySQL+PHP

知识讲解

PHP工作原理

知识讲解

PHP环境搭建

Technology Tarena 达内科技

- 1.安装apache服务器
- 2.安装PHP(只需要将压缩包加压到相应的位置即可)
- 3.PHP配置，将php.ini-dist更名为php.ini，当成PHP的配置文件
- 4.配置Apache，修改apache中conf目录中的httpd.conf配置文件
  在配置文件中添加以下几句：
- 1》加载PHP模块：LoadModule php5_module
  “php5apache2_2.dll核心文件所在位置及名称”
- 2》PHP配置文件所在位置：PHPIniDir “PHP配置文件所在位置”
- 3》哪些类型的文件将交由PHP引擎处理:
- AddType application/x-httpd-php .php
- 5.保存之后重新启动服务器

知识讲解

PHP文档组成

Technology Tarena 达内科技

HTML/XHTML标记

服务器端脚本（PHP脚本）

客户端脚本(如JavaScript、jQuery等)

某个文档是否以.php为扩展名，取决于是否存在服务器端脚本。

知识讲解

Zend Studio简介

Technology Tarena 达内科技

- 6.创建带有标准HTML结构的PHP文档
- A、Window(窗口)菜单 --> Preferences(参数设置)
- B、General(基本) --> Content Types(内容类型) --> Text --> HTML
- C、单击“Add(添加)”按钮，在弹出的对话框中输出“*.php”
- D、单击“OK”按钮

知识讲解

PHP语法结构

Technology Tarena 达内科技

XML风格

```
<?php
    …
    …
?>
```

注：对于只包含有 PHP 代码的文件，结束标志是不允许存在的。

```
1 <?php
2 //XML风格
3 echo "这是XML风格";
4 ?>
```

PHP语法结构

短风格

<?

...

...

?>

注：需要启用PHP配置文件的short_open_tag指令。

```
1 <?
2 //短风格 --必须开启PHP配置文件的short_open_tag选项
3 echo "这是短风格";
```

PHP语法结构

ASP风格

<%

…

…

%>

注：需要启用PHP配置文件的asp_tags指令。

```
1 <%
2 //ASP风格--必须开启PHP配置文件的asp_tags选项
3 echo "这是ASP风格";
4 %>
```

PHP语法结构

Script风格(脚本风格)

<script language="php">

…

…

</script>

```
1 <script language="php">
2 //Script风格
3 echo "这是Script风格";
4 </script>
```

知识讲解

PHP注释

单行注释（C++风格）

// 注释内容

```
<?php
//文件名称
$filename = '3kd3kx3.jpg';
?>
```

知识讲解

PHP注释

单行注释（Shell风格）

注释内容

```
<?php
#年龄
$age = 27;
?>
```

PHP注释

多行注释（C　　风格）

```
/*
    注释内容
    ...
*/
```

```
/*
$i = 7;
echo "This is a test";
*/
```

注:使用多行注释时，不能让注释陷入递归循环。

代码实践

课堂练习

Technology Tarena 达内科技

输出Hello World

```
<?php
echo 'Hello World!';
?>
```

Technology Tarena 达内科技

“

变量

”

知识导入

变量概述

变量是命名的内存位置，其中包含数据，可以在程序执行期间进行处理。

知识讲解

声明变量

Technology Tarena 达内科技

$变量名称;

$变量名称,$变量名称,...;

$变量名称 = 变量值;

说明:由于PHP是“弱”语言，其对于变量声明没有特殊要求，也就是说变量既可以先声明，后使用；也可以直接使用，而无需声明。

变量赋值

$变量名称 = 变量值;

$变量名称=$变量名称...=变量值

知识讲解

声明变量

Technology Tarena 达内科技

声明变量需要注意：
A、PHP变量必须以美元符号($)为前缀

B、变量名称必须以字母或下划线开头，包含字母、数字及下划线

C、变量名称禁止包含空格、斜线、反斜线等特殊符号

D、变量名称必须含义明确

E、变量名称必须遵守“驼峰标记法”，包括大驼峰和小驼峰
小驼峰：firstName
大驼峰：LastName

Technology Tarena 达内科技

“常量”

常量概述

一个简单值的标识符,在脚本执行期间该值不能改变。

知识讲解

定义常量

Technology Tarena 达内科技

define(string name,mixed value);

常量的数据类型只能为标量类型。

常量的名称一般由大写字母组成。

常量在定义和引用时无需美元符号。

```
<?php
define('MIN_VALUE', '0.0');
define('MAX_VALUE', '1.0');
echo MIN_VALUE; //0.0
echo MAX_VALUE;  //1.0
```

知识讲解

魔术常量

__LINE__
说明:文件中的当前行号
__FILE__
说明:文件的完整路径和文件名
__CLASS__
说明:类的名称
__METHOD__
说明:类的方法名称
__FUNCTION__
说明:函数名称

Technology Tarena 达内科技

“

输出语句

”

知识讲解

输出语句

Technology Tarena 达内科技

echo

描述:输出一个或多个字符

语法:void echo(string $arg[,string $arg[,...]])

print

描述:打印字符

语法: mixed print(string $arg)

输出语句

print_r

描述:打印变量易于理解的信息

语法: mixed print_r(mixed $expr)

var_dump

描述:打印变量信息

语法: void var_dump(mixed $expr[,...])

Technology Tarena 达内科技

“数据类型”

数据类型概述

数据类型是具有相同特性的一组数据的统称。

PHP支持8种基本数据类型(Primitive Type)和3种伪类型（Pseudo-types）。

知识讲解

基本数据类型

Technology Tarena 达内科技

标量类型(Scalar Type)
复合类型(Compound Type)
特殊类型(Special Type)

知识讲解

标量类型

Technology Tarena 达内科技

标量类型只能包含单一信息，包括：

- 字符型（String）
- 布尔型(Boolean)
- 整数型(Integer)
- 浮点型(Float)

知识讲解

字符型

- 字符型数据必须括在定界符之间。
  - 单引号 ''
  - 双引号 ""
  - 定界符 <<<

知识讲解

单引号

```
<?php
$username =  ‘Mr.King';
$password =  ‘123456';
```

双引号

```
<?php
$username =  “Mr.King";
$password =  “123456";
```

知识讲解

Heredoc

<<<标记名称
…
…
…
标记名称;

```
$str = <<<EOF
        <html>
                <head><title>标题</title></head>
                <body>
                        <p align="center">Mr.King</p>
                </body>
        </html>
EOF;
echo $str;
```

- 结束标记必须位于行首，不能有缩进和空格，且在结束标记末尾要有分号 。
- 开始标记和开始标记相同。

知识讲解

转义符

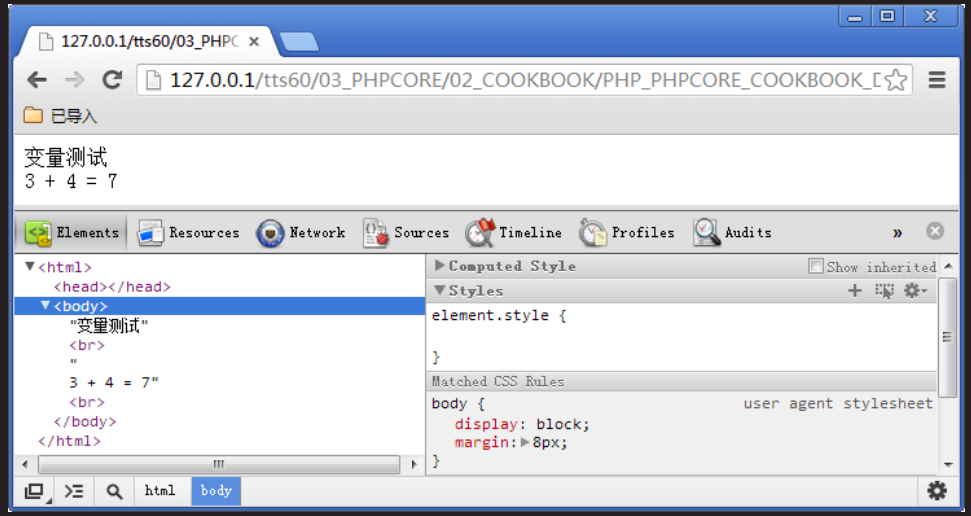
转义符	描述
\n	换行
\r	回车
\t	水平制表符
\v	垂直制表符
\f	换页符
\\	反斜线
\$	美元符号
\’	单引号
\”	双引号

知识讲解

单引号与双引号的区别

双引号内的转义符或变量可以被PHP引擎正确解析；而单引号仅识别\’和\\。

```
<?php
$i = 3;
$n = 4;
echo “变量测试<br/>\n”;
echo “{$i} + {$n} = ” . ($i+ $n) . “<br/>\n”;
```



知识讲解

转义符与HTML实体

Technology Tarena 达内科技

如果输出的字符将直接打印到浏览器，那么双引号/单引号就必须采用HTML实体表示；如果双引号/单引号将在浏览器的源代码中出现，那么就必须采用转义符。

```
<?php
$memory = '&quot;A&gt;B&quot;';
$tag = "<p align=\"center\">这是段落</p>";
```

知识讲解

布尔型

布尔型（Boolean）使用关键字true/false或TRUE/FALSE表示。

```
<?php
$sex = true;
$merried = FALSE;
```

整型

整型可以用十进制，十六进制或八进制表示，可添加符号位。
如果用八进制表示，数字前必须添加0。
如果用十六进制表示，数字前必须添加0X/0x。
注：

- PHP 不支持无符号整数
- 数字如果超出 integer 的范围，将会被解释为 float

```
$i  = 23;       //十进制，23
$n  = 035;      //八进制，转换成十进制为29
$m =  0XF12;  //十六进制，转换成十六进制为3858
```

知识讲解

浮点型

浮点型（Float）数据的字长和平台相关。
通常最大值是1.8e308并具有14位十进制数字的精度。

```
<?php
$salary = 4589.75;
echo gettype($salary); //输出double
```

知识讲解

复合类型

Technology Tarena 达内科技

允许将多个相同类型的项聚集起来，表示为一个实体。包括：

- 数组(Array)
- 对象(Object)

```
<?php
$arr = array(34,67,8,25); //数组
$obj = new stdClass();    //对象
echo gettype($arr);       //输出array
echo gettype($obj);       //输出object
```

知识讲解

特殊类型

Technology Tarena 达内科技

提供某种特殊用途的类型，无法归入其他任何类型。包括：

- 资源(Resource)，一种特殊的数据类型，用来表示一种PHP的外部资源，例如数据库访问，文件访问、目录操作、图像操作等。程序员将永远无法触及这种类型的变量，必须通过专门的函数访问。
- 空值(NULL)
  - 变量赋值为NULL
  - 没有赋值的变量
  - 使用unset销毁的变量

知识讲解

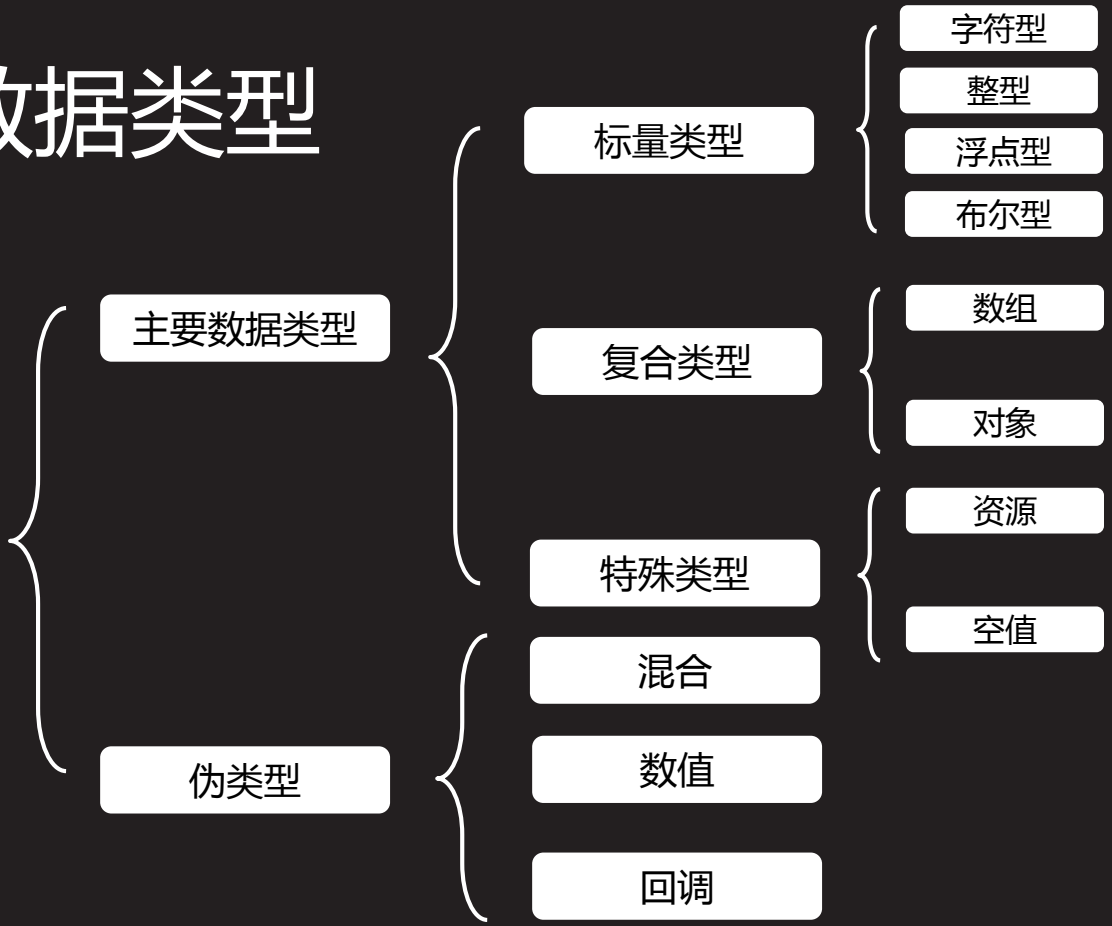
伪类型

Technology Tarena 达内科技

- mixed
  混合型数据可以接受多种不同的（但并不必须是所有的）类型。
- number
  数值型可以是 integer 或者 float。
- callback
  回调函数以接受用户自定义的函数作为参数。还可以是一个对象的方法，包括静态类的方法。

知识讲解

数据类型



知识讲解

数据类型转换

自动转换(隐式转换)

强制转换(显式转换)

知识讲解

转换成布尔型

Technology Tarena 达内科技

当转换为布尔型时，以下值被认为false:

- 布尔值false
- 整型0
- 浮点型0.0
- 空白字符串或字符串'0’
- 没有成员变量的数组
- 特殊类型NULL

所有其他值都被认为true(包含任何资源类型)

知识讲解

转换成数值型

Technology Tarena 达内科技

- 布尔型false转换为0，true转换为1。
- 浮点数转换成整数时，数字将被取整。
- 字符串如果包括“.”，“e”或“E”其中任何一个字符的话，字符串被当作 float 来求值。否则就被当作整数。
- 如果字符串以合法的数字数据开始，就用该数字作为其值，否则其值为 0（零）。

转换成数值型


```php
<?php
$i = 3 + true; //4
$n = '3.5str' + 4; //7.5
$m = '2a6' * 2; // 4
$t = 's36' + 5; //5
```

知识讲解

转换成字符型

Technology Tarena 达内科技

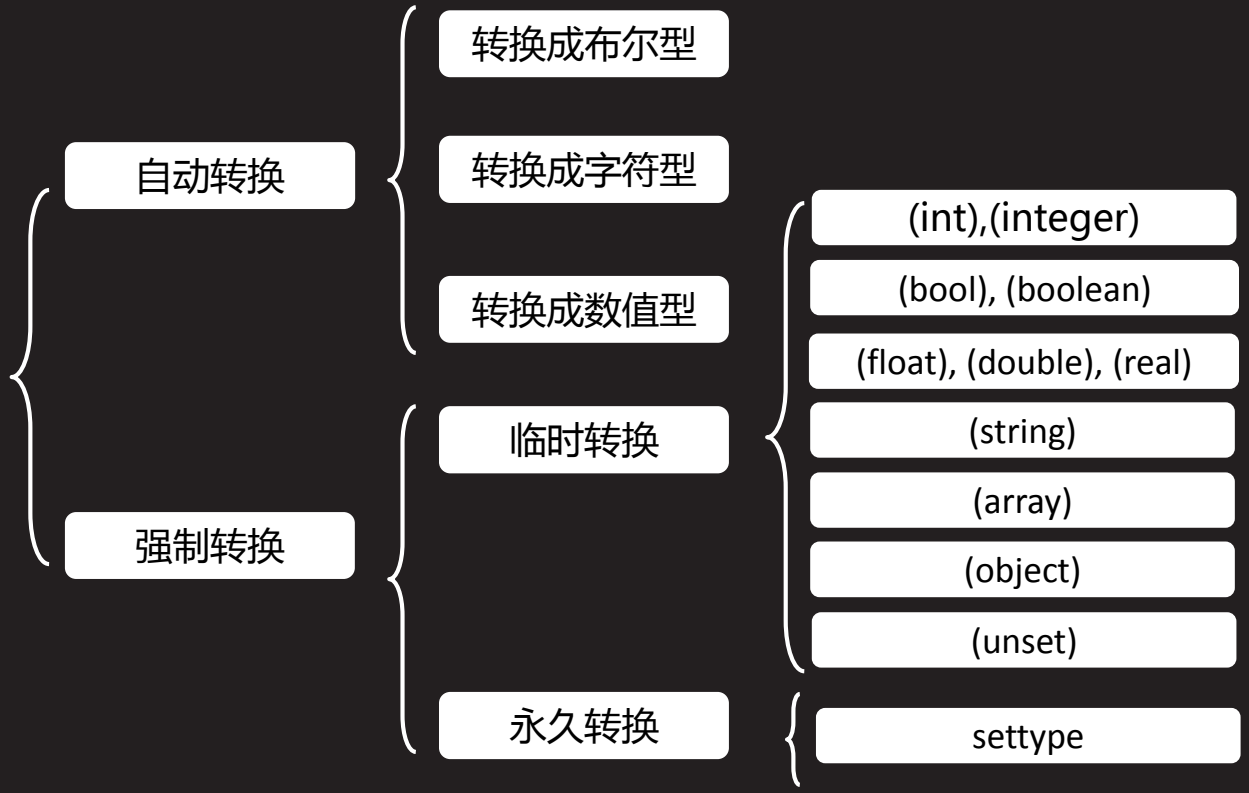
- 布尔值 TRUE 将被转换为字符串 "1"，而值 FALSE 将被表示为空字符串
- 整数或浮点数数值在转换成字符串时，字符串即为数字本身
- 数组将被转换成字符串Array
- 对象将被转换成字符串 Object
- 资源类型将会以 "Resource id #序号" 的格式被转换成字符串
- NULL 将被转换成空字符串

知识讲解

显示转换

(int), (integer) – 转换成整型
(bool), (boolean) – 转换成布尔型
(float), (double), (real) - 转换成浮点型
(string) - 转换成字符型
(array) - 转换成数组
(object) - 转换成对象
(unset) – 转换成空值 (PHP 5)

数据类型转换


自动转换
强制转换
转换成布尔型
转换成字符型
转换成数值型
临时转换
永久转换
(int),(integer)
(bool), (boolean)
(float), (double), (real)
(string)
(array)
(object)
(unset)
settype

Technology Tarena 达内科技

“

运算符

”

知识讲解

运算符

- 算数运算符
- 字符运算符
- 赋值运算符
- 比较运算符
- 逻辑运算符
- 位运算符
- 其它运算符
- 运算符的优先级

运算符

PHP中的运算符

运算符是可以通过给出的一或多个值（用编程行话来说，表达式）来产生另一个值（因而整个结构成为一个表达式）的东西。所以可以认为函数或任何会返回一个值（例如 print）的结构是运算符，而那些没有返回值的（例如 echo）是别的东西。

有三种类型的运算符：

- 一元运算符，只运算一个值，例如！（取反运算符）或 ++（加一运算符）。
- 二元运算符，有两个操作数，PHP支持的大多数运算符都是这种。
- 三元运算符：? :。它应该被用来根据一个表达式在另两个表达式中选择一个，而不是用来在两个语句或者程序路线中选择。把整个三元表达式放在扩号里是个很好的主意。

知识讲解

算术运算符

Technology Tarena 达内科技

运算符	意义	示例	结果
+	加法运算	$a+$b	$a和$b的和
-	减法/取负运算	$a-$b	$a和$b的差
*	乘法运算	$a*$b	$a和$b的积
/	除法运算	$a/$b	$a和$b的商
%	求余运算符（取模运算）	$a%$b	$a和$b的余数
++	累加1	$a++或++$a	$a的值加1
--	递减1	$a--或--$a	$a的值减1

知识讲解

字符串运算符

Technology Tarena 达内科技

- 字符串运算符比较简单，用连接符号点 .

知识讲解

赋值运算符

Technology Tarena 达内科技

基本的赋值运算符是“=”。一开始可能会以为它是“等于”，其实不是的。它实际上意味着把右边表达式的值赋给左边的运算数。

运算符	意义	示例
=	将一个值或表达式的结果赋给变量	$x=3
+=	将变量与所赋的值相加后的结果赋给该变量	$x+=3等价于$x=$x+3
-=	将变量与所赋的值相减后的结果赋给该变量	$x-=3等价于$x=$x-3
*=	将变量与所赋的值相乘后的结果赋给该变量	$x*=3等价于$x=$x*3
/=	将变量与所赋的值相除后的结果赋给该变量	$x/=3等价于$x=$x/3
%=	将变量与所赋的值求模后的结果赋给该变量	$x%=3等价于$x=$x%3
.=	将变量与所赋的值相连后的结果赋给该变量	$x.="H"等价于$x=$x."H"

知识讲解

比较运算符

Technology Tarena 达内科技

运算符	描述	说明	示例
>	大于	当左边大于右边时返回true，否则返回false	$a>$b
<	小于	当左边小于右边时返回true，否则返回false	$a<$b
>=	大于等于	当左边大于等于右边时返回true，否则false	$a>=$b
<=	小于等于	当左边小于等于右边时返回true，否则false	$a<=$b
==	等于	两边操作数的值相等时返回true，否则false	$a==$b
===	全等于	两边值相等并且类型相等返回true，否则false	$a===$b
<> 或!=	不等于	两边值不等时返回true，否则返回false	$a<>$b $a!=$b
!==	非全等于	两边值与类型都相同时返回false，否则true	$a!==$b

知识讲解

逻辑运算符

Technology Tarena 达内科技

运算符	描述	说明	示例
and或&&	逻辑与	当两边操作数都为true时，返回true，否则返回false	$a and $b $a && $b
or或||	逻辑或	当两边操作数都为false时，返回false，否则返回true	$a or $b $a || $b
not或！	逻辑非	当操作数为true时返回false，否则返回true	not $b !$b
xor	逻辑异或	当两边操作数只有一个为true时，返回true，否则返回false	$a xor $b

知识讲解

位运算符

Technology Tarena 达内科技

运算符	描述	说明	示例
&	按位与	只有参与运算的两位都为1时，运算结果才为1，否则为0.	$a & $b
|	按位或	只有参与运算的两位都为0时，运算结果才为0，否则为1.	$a | $b
^	按位异或	只有参与运算的两位不同，运算结果才为1，否则为0.	$a^$b
~	按位非	将用二进制表示的操作数中的1变成0，0变成1.	~$a
<<	左移	将左边的操作数在内存中的二进制数据右移右边操作数指定的位数，右边移空的部分补上0	$a<<$b
>>	右移	将左边的操作数在内存中的二进制数据左移右边操作数指定的位数，左边移空的部分补上0	$a>>$b

知识讲解

其它运算符

Technology Tarena 达内科技

运算符	描述	示例
? :	三元运算符，可以提供简单的逻辑判断。	$a<$b?$c=1:$c=0
``	反引号(` `)是执行运算符，PHP将尝试将反引号中的内容作外壳命令来执行，并将其输入信息返回	$a=`ls -al`
@	错误控制运算符，当将其放置在一个PHP表达式之前，该表达式可能产生的任何错误信息都被忽略掉。	@表达式
=>	数组下标指定符号，通过此符号指定数组的键与值。	键=>值
->	对象成员访问符号，访问对象中的成员属性或成员方法。	对象->成员
instanceof	类型运算符，用来测定一个给定的对象是否来自指定的对象类。	对象instanceof 类名

知识讲解

表达式

Technology Tarena 达内科技

- 表达式是 PHP 最重要的基石。在 PHP 中，几乎所写的任何东西都是一个表达式。简单但却最精确的定义一个表达式的方式就是“任何有值的东西”。
- 最基本的表达式形式是常量和变量。当键入“$a = 5”，即将值“5”分配给变量 $a。“5”，很明显，其值为 5，换句话说“5”是一个值为 5 的表达式（在这里，“5”是一个整型常量）。
- 赋值之后，所期待情况是 $a 的值为 5，因而如果写下 $b = $a，期望的是它犹如 $b = 5 一样。换句话说，$a 是一个值也为 5 的表达式。如果一切运行正确，那这正是将要发生的正确结果。
- 稍微复杂的表达式例子就是函数。

知识讲解

运算符优先级

Technology Tarena 达内科技

结合方向	运算符	附加信息
无	clone new	clone 和 new
左	[	array()
右	++ -- ~ (int) (float) (string) (array) (object) (bool) @	类型和递增／递减
无	instanceof	类型
右	!	逻辑运算符
左	* / %	算术运算符
左	+ - .	算术运算符和字符串运算符
左	<< >>	位运算符
无	== != === !== <>	比较运算符
左	&	位运算符和引用
左	^	位运算符
左	|	位运算符
左	&&	逻辑运算符
左	||	逻辑运算符
左	? :	三元运算符
右	= += -= *= /= .= %= &= |= ^= <<= >>= =>	赋值运算符
左	and	逻辑运算符
左	xor	逻辑运算符
左	or	逻辑运算符
左	,	多处用到

知识讲解

PHP中的预定义变量

Technology Tarena 达内科技

- $_SERVER:服务器变量
- $_ENV:环境变量
- $_GET:HTTP GET 变量
- $_POST:HTTP POST变量
- $_FILES:HTTP 文件上传变量
- $_COOKIE:HTTP Cookies
- $_SESSION:Session 变量
- $_REQUEST:HTTP Request变量，默认包括$_GET+$_POST+$_COOKIE
- $GLOBALS:包含了全部变量的全局组合数组。

知识讲解

流程控制

- 任何 PHP 脚本都是由一系列语句构成的。一条语句可以是一个赋值语句，一个函数调用，一个循环，一个条件语句或者甚至是一个什么也不做的语句（空语句）。语句通常以分号结束。此外，还可以用花括号将一组语句封装成一个语句组。语句组本身可以当作是一行语句。
- 顺序结构：程序会按照自上而下的顺序执行。
- 分支结构：if … if …else … if elseif .. switch
- 循环结构：for … while … do while

知识讲解

分支结构

Technology Tarena 达内科技

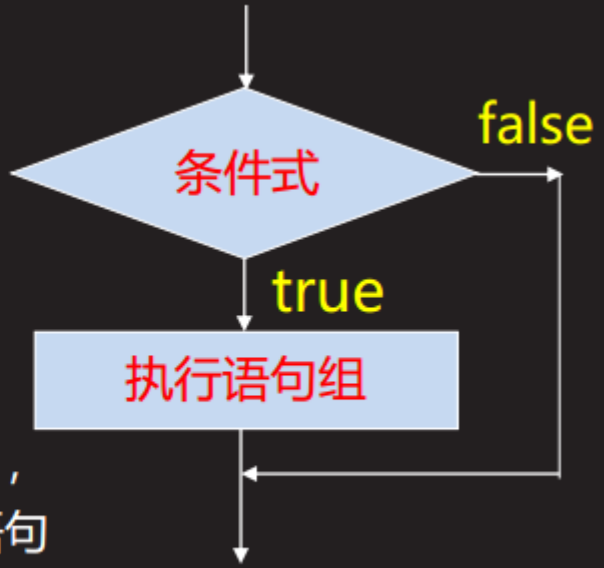
- 单一条件分支结构(if)
- 双向条件分支结构(else从句)
- 多向条件分支结构(elseif子句)
- 多想条件分支结构(switch语句)等价于廉价的if elseif
- 巢状条件分支结构

单一条件分支结构

基本语法：

```
if(条件表达式){
        语句组;
        //语句组为单条语句时可省略“{}”。
}
```

当条件表达式的值为真（true）时，PHP将执行语句组，相反条件表达式的值为假（false）时，PHP将不执行语句组，忽略语句组执行下面的语句。

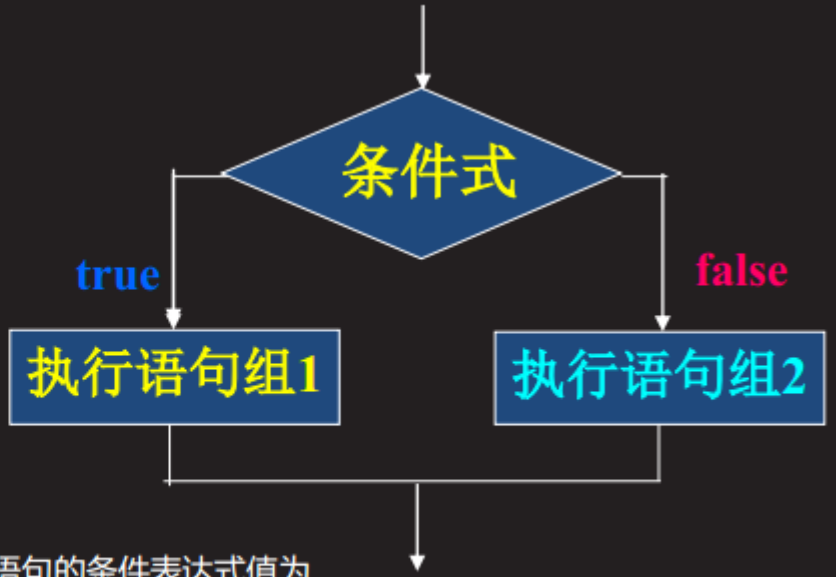


双向条件分支结构

if...else语句：

格式如下

```
if(条件表达式){
        语句组1
}else{
        语句组2
        //语句组为单条语句时可省略“{}”。
}
```



if-else 条件判断与 if 条件判断类似，所不同的是，if-else 语句的条件表达式值为真(true)时，会执行 if 的本体语句(语句组1)，而条件表达式值为假(false)时，则执行 else 的本体语句(语句组2)。

知识讲解

多向条件分支结构

语法格式如下：

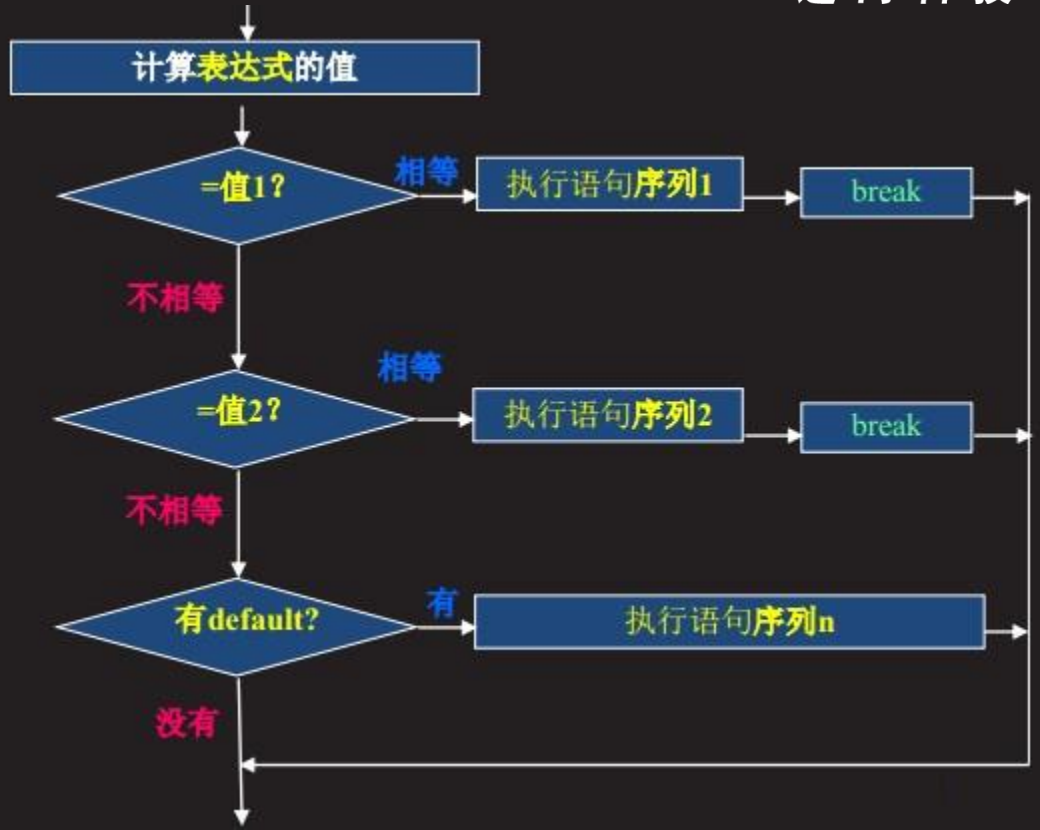
```
if(条件表达式1){
        语句块1
}elseif(条件表达式1){
        语句块2
... ...
}elseif(条件表达式n){
        语句块n
}else{
        语句块n+1
}
```

知识讲解

多向条件分支结构

switch- case语句语法：

```
switch ( 表达式 ) {
        case  值1 :
                语句序列1 ;
                break ;
        case  值2 :
                语句序列2 ;
                 break ;
        ...
        default :
                语句序列 n ;
                break ;
}
```



知识讲解

Switch语句

Technology Tarena 达内科技

当程序执行碰到switch条件判断时，它会取出键值，然后与语句体中的case所列出的值逐一比较，如果数值不符合，则将数值往下一个case传递，如果数值符合，则执行case中的语句，然后再碰到break语句即跳出switch条件判断，如果所有的值比对都不符合，则会执行default中的语句。

switch语句使用注意事项:

- switch语句与if语句不同，它仅能判断一种关系：是否恒等。
- switch语句中case子句的常量可以是整型常量、字符型常量、表达式或变量。
- 在同一个switch中，case子句的常量不能相同，否则第二个值永远无法匹配到。
- case 和 default 子句后面的语句序列允许由多个可执行语句组成，且不必用 “{ }” 括起来，也可以为空语句。
- switch语句中可省略break语句和default子句。但省略后会改变流程。

知识讲解

巢状条件分支结构

语法:

```
if(表达式1){
   if(表达式2){
                  .. .. ..
               }else{
     .. .. ..
                  }
}else{
   if(表达式3){
   ...
                  }
}
```

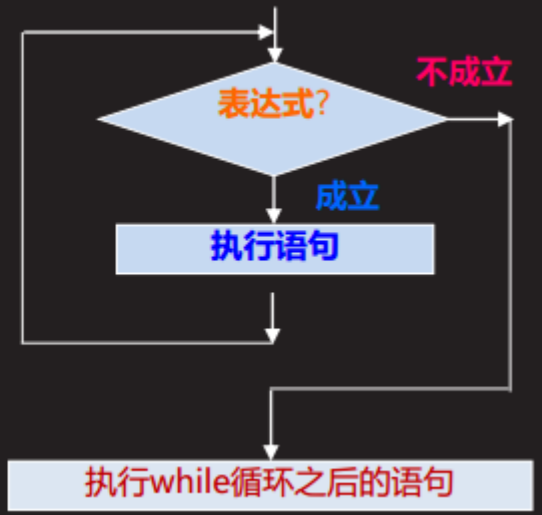
- 巢状式条件分支结构就是if语句的嵌套，即指if或else后面的语句块中又包含if语句。if语句可以无限层地嵌套在其他if语句，这给程序的不同部分的条件执行提供了充分的弹性。

循环结构-while循环

- while循环语法：

```
while(表达式){
        语句或语句序列
        ... ...
}
```

- 当while循环语句中表达式（循环控制语句）的结果为真时，程序将反复执行同一段程序：循环体（ while中的语句或语句序列），直到满足一定的条件（表达式的结果为假时）后才停止执行该段程序。

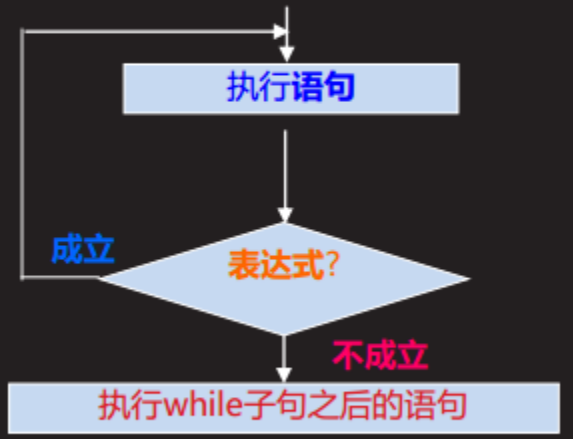


循环结构-do...while循环

- 基本格式：

```
do{
        语句或语句序列
        ... ...
}while(表达式);
```

- 程序会先执行 do 语句体中的语句（循环体），然后再检查表达式（循环控制语句）的值，如果符合条件式（值为真），就再进行 do 语句体中的语句，直到条件不符合为止。

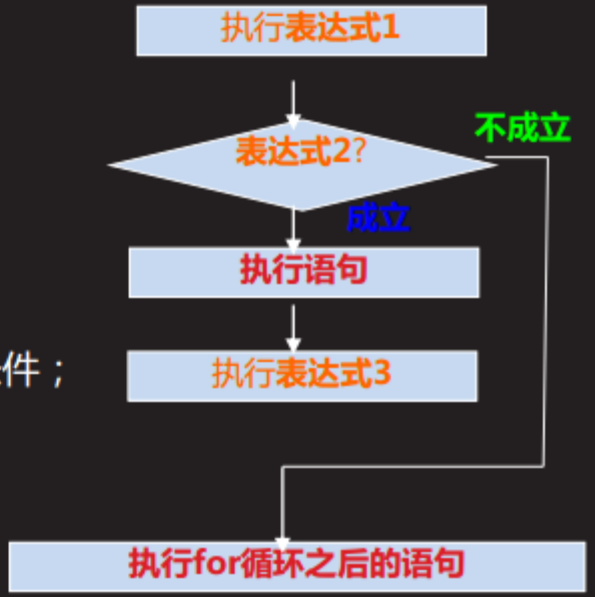


循环结构-do...while循环

- 基本格式：

```
for(表达式1;表达式2;表达式3)
{
        语句或语句序列;
}
```

- for循环语句中表达式1为循环初始条件；表达式2为循环控制条件；表达式3为控制变量递增；语句或语句序列为循环体。



知识讲解

特殊流程控制语句

Technology Tarena 达内科技

1. break

- 我们之前在 switch 条件判断中已经使用过 break 关键字，它会使得程序流程离开 switch 本体中的语句，如果 break 使用在 for、while 或 do-while 循环结构中时，将会使得程序离开该层循环。

2. continue

- continue 的作用与 break 有点类似，continue 若使用在 for、while 或 do-while循环结构中，当程序执行至 continue 时，之后的语句将直接被略过，而直接执行下一次的循环动作。

3. exit

- 当前的脚本中只要执行到exit语句，而不管它在哪个结构中都会直接退出当前脚本。exit是一个函数，当前使用过的die( )函数就是exit( )的别名。可以带参数输出一条消息，并退出当前脚本。

知识讲解

数组

Technology Tarena 达内科技

- 1.什么是数组？数组是变量的集合
- 2.数组类型：索引数组和关联数组
- 索引数组：下标为数字
- 关联数组：下标为字符
- 3.数组的声明：
- $数组名称=array();//空数组
- $数组名称=array(值,值,值...);//索引数组，下标连续
- $数组名称=array(下标=>”值”,下标=>”值”...)如果下标为字符则为关联数组，下标为数字则为索引数组，既有数字又有字符为混合数组
- $变量名称[]=值;或者$变量名称[数字]=值;或者$变量名称[字符]=值;
- Range():快速创建数组
- Compact():快速创建数组

知识讲解

数组使用注意事项

Technology Tarena 达内科技

- 1.如果没有指定编号的索引数组，下标从0开始编号
- 2.如果指定编号，则索引数组按指定的编号顺序
- 3.如果没有指定下标的成员前的其它成员的下标为正数，则该成员的下标为已有最大下标+1；
- 4.如果没有指定下标的成员前的其它成员下标都为负数，那么该成员的下标为0
- 5.如果某成员钱的成员为关联数组，那么该成员在没有指定下标的情况下，其下标为0

知识讲解

数组的使用

- 数组中的值通过下标来取
- $数组名称[下标]

知识讲解

数组的遍历

Technology Tarena 达内科技

- 数组通过foreach语句遍历数组
- 1.foreach($array as $value){
- …
- }
- 2.foreach($array as $key=>$value){
  …
- }

知识讲解

PHP函数

Technology Tarena 达内科技

- 1.什么是函数？函数(Function)就是可以完成某种特定功能的代码段，函数由函数名和函数体组成。
- 2.函数的语法结构
- Function 函数名称 ([$参数名称[=值]][,...]){
  函数体
  return 返回值
- }
- 注意：
- 1》函数名称必须以字母或者下划线开头，包含字母、数字、下划线，禁止包含特殊字符
- 2》函数名称最好以动词开头
- 3》函数名称最好含义明确
- 4》函数名称最好遵循驼峰标记法
- 5》函数可以有零个或者多个参数

知识讲解

PHP函数-参数

Technology Tarena 达内科技

- 函数的参数可以有也可以没有，如果指定了参数，在调用函数的时候必须传递参数，如果指定了可选参数，调用的时候可以不传参。
- 注意：
- 1》函数可以有零个或者多个参数
- 2》函数的参数可以为任意类型，除了资源
- 3》有默认值的参数称为可选参数，没有默认值的参数称为必选参数
- 4》必选参数在函数调用的时候必须传
- 5》所有的必选参数必须位于所有可选参数之前

知识讲解

PHP函数的调用

Technology Tarena 达内科技

- 函数定义完之后不调用不起作用
- 函数通过以下方式调用：
- [$变量名称=]函数名称([参数值[,...]]);

知识讲解

PHP函数的返回值

- 函数的返回值通过return语句实现
- 1.默认情况下，函数的返回值为NULL
- 2.函数碰到return语句将终止函数的执行
- 3.返回值可以为任意数据类型

知识讲解

PHP函数的执行流程

- 函数声明之后不调用不被执行，首先声明之后先将函数调入计算机内存，在调用函数的时候，PHP将控制权移交到相关函数的函数体，之后开始执行函数体内的代码，当明确遇到return语句或函数体执行完毕，再将控制权移交到调用该函数的那行代码。

知识讲解

PHP变量的作用域

Technology Tarena 达内科技

- 1.PHP变量的作用域是从声明处开始到当前脚本结束的位置
- 2.局部变量：函数体中声明的变量则为局部变量，在函数调用的时候被声明，函数执行结束后被释放
- 3.静态变量：静态变量也是存在与函数体内，通过关键字static声明的变量为静态变量，静态变量保存在静态内存中，函数执行结束后静态变量不释放，而是保存在静态内存中，当再次调用函数的时候在从静态内存中取出来。
- 4.全局变量：函数体外声明的变量都为全局变量。函数体内的局部变量会覆盖全局变量，所以在函数体内如果想使用函数外的变量需要使用global关键字定义目标变量，告诉函数体此变量为全局变量。也可以使用$GLOBALS全局变量数据去调用变量。

PHP函数的使用

- 1.变量函数：如果将函数名称赋予变量，并且在调用变量时，如果带有小括号，那么PHP引擎将尝试解析为函数。
- 2.递归函数：递归函数指函数调用自身。
- 3.回调函数：如果调用函数的时候传参传的是另一个函数的函数名称，并且在函数体内通过变量函数的形式进行了调用。